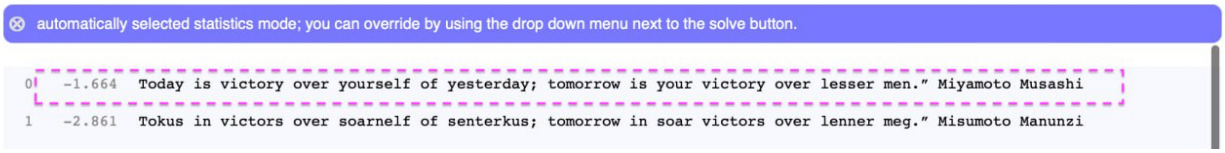
تسک ۱:

سوال ۱:

با استفاده از quipquip **داریم:**



تسک ۲:

سوال ۱:

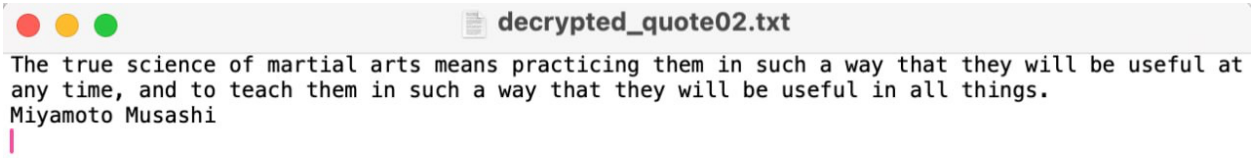
```
(base) parnian@parnians-MacBook-Pro task02 % gpg --batch --passphrase 's!kR3T55'
 --cipher-algo AES256 --decrypt quote01.txt.gpg
gpg: AES256.CFB encrypted data
gpg: encrypted with 1 passphrase
Do not waste time idling or thinking after you have set your goals.
Miyamoto Musashi
```

این دستور از ابزار GPG **برای رمزگشایی فایل** quote۰۱.txt.gpg **استفاده می‌کند. سوئیچ** --batch **اجازه می‌دهد که فرمان به صورت غیرتعاملی اجرا شود، و** --passphrase 's!kR۳۵۵' **رمز عبور لازم برای رمزگشایی را فراهم می‌کند. سوئیچ** --cipher-algo AES۲۵۶ **الگوریتم رمزنگاری** AES۲۵۶ **را تعیین می‌کند و** --decrypt **مشخص می‌کند که عملیات باید رمزگشایی**

باشد. این دستور در نهایت فایل ورودی را رمزگشایی و به صورت متنی برگردانده یا در خروجی ذخیره می‌کند.

سوال ۲:

```
(base) parnian@parnians-MacBook-Pro task02 % openssl aes-256-cbc -d -in quote02
-out decrypted_quote02.txt -pass pass:'s!kR3T55'
```



این دستور از ابزار OpenSSL برای رمزگشایی فایل quote۰۲ استفاده می‌کند. سوئیچ -d به معنای عملیات رمزگشایی است، -in فایل ورودی رمزگذاری‌شده، -out فایل خروجی رمزگشایی‌شده (یعنی decrypted_quote۰۲.txt)، و -pass pass:'s! kR۳T۵۵' رمز عبور برای فرآیند رمزگشایی را مشخص می‌کند. الگوریتم رمزنگاری استفاده‌شده AES-۲۵۶-CBC است.

سوال ۳:

```
(base) parnian@parnians-MacBook-Pro task02 % gpg --batch --passphrase 's!kR3T55'
 --cipher-algo CAMELLIA256 --decrypt quote03.txt.gpg
gpg: CAMELLIA256.CFB encrypted data
gpg: encrypted with 1 passphrase
You must understand that there is more than one path to the top of the mountain.
Miyamoto Musashi
```

این دستور از ابزار GPG برای رمزگشایی فایل quote۰۳.txt.gpg استفاده می‌کند. سوئیچ --batch عملیات را به صورت غیرتعاملی اجرا می‌کند، --passphrase 's! kR۳T۵۵' رمز عبور برای رمزگشایی را تعیین می‌کند، --cipher-algo CAMELLIA۲۵۶ الگوریتم رمزنگاری Camellia-۲۵۶ را برای رمزگشایی مشخص می‌کند، و --decrypt دستور به GPG می‌دهد که فایل ورودی را رمزگشایی کند.

تسک ۳:

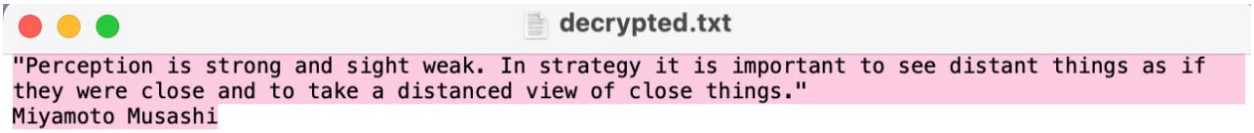
سوال ۱:

```
(base) parnian@parnians-MacBook-Pro task03 % openssl pkeyutl -decrypt -in
 ciphertext_message -inkey private-key-bob.pem -out decrypted.txt
```

دستور openssl pkeyutl -decrypt -in ciphertext message -inkey private-key-bob.pem -out decrypted.txt از ابزار OpenSSL برای رمزگشایی یک پیام رمزنگاری‌شده

استفاده می‌کند. سوئیچ -decrypt عملیات رمزگشایی را مشخص می‌کند، -in ciphertext message فایل ورودی رمزنگاری‌شده را تعیین می‌کند، -inkey private-key-bob.pem کلید خصوصی مورد استفاده برای رمزگشایی (در اینجا private-key-bob.pem) را مشخص می‌کند، و -out decrypted.txt فایل خروجی که پیام رمزگشایی‌شده در آن ذخیره می‌شود را تعیین می‌کند.

متن اصلی به صورت زیر است:



سوال ۲و۳:

```
(base) parnian@parnians-MacBook-Pro task03 % openssl rsa -in private-key-bob.pem
 -text -noout
```

این دستور برای نمایش جزئیات یک کلید خصوصی RSA استفاده می‌کند. سوئیچ -in private-key-bob.pem فایل ورودی کلید خصوصی را مشخص می‌کند، -text اطلاعات و جزئیات کلید (مانند مولفه‌های مختلف RSA: مدول، توان عمومی و خصوصی و غیره) را به صورت متنی نمایش می‌دهد، و -noout مانع از نمایش خود کلید در خروجی می‌شود، به‌طوری‌که فقط جزئیات کلید نمایش داده شود.

```
prime1:
    00:ff:ea:65:3e:e5:96:96:0b:66:55:f1:f9:d0:37:
    66:e9:35:a5:c3:43:ca:66:75:40:49:46:8d:85:a7:
    ff:f4:73:97:69:11:a1:1e:37:f9:e3:38:cb:c0:5e:
    56:e9:1a:0d:f2:9f:80:56:87:2a:99:bb:88:8e:93:
    35:5a:9a:c6:f7:99:44:90:88:09:33:a6:0d:ea:b4:
    56:98:66:20:9c:34:e7:b9:33:64:4f:08:01:08:62:
    44:68:8f:df:79:0d:84:2b:77:e7:03:8b:3c:7a:e3:
    e0:e0:ee:23:64:22:51:ed:dd:b8:1c:b3:75:c4:3f:
    4a:cf:fc:7c:57:0b:95:75:e7
prime2:
    00:e8:72:11:5c:b5:5c:14:19:85:ce:e7:d2:e9:54:
    7b:58:ae:32:e9:e6:39:a7:65:b4:90:2f:53:b5:9d:
    22:62:84:fe:52:86:f5:01:a2:9c:b0:4f:80:ee:d4:
    07:27:3b:69:02:70:33:da:7d:97:56:b9:3e:f3:a1:
    84:9e:73:6a:47:e5:99:8c:44:86:75:c1:bf:71:89:
    06:b0:ee:dd:16:45:e7:05:fa:02:bd:e6:3e:b7:f2:
    fe:e7:22:0b:ed:ca:23:a0:68:0b:fe:fb:c3:57:19:
    21:58:6e:73:1d:9d:3c:2a:8a:c1:7e:ea:73:67:5a:
    cb:3d:a8:9b:be:50:08:9e:27
```

تسک ۴:

سوال ۱:

```
(base) parnian@parnians-MacBook-Pro task04 % openssl dhparam -in dhparams.pem -t
ext -noout
    DH Parameters: (4096 bit)
        prime:
            00:c0:10:65:c6:ad:ed:88:04:88:1e:e7:50:1b:30:
            0f:05:2c:2d:d4:ea:60:44:9e:2a:f7:90:02:89:a4:
            7e:05:99:32:38:dc:75:50:0a:c7:f6:6b:f7:b4:9a:
            df:ef:ca:e0:ce:55:5d:31:48:3e:9c:35:5a:ad:03:
            9c:87:d7:1c:48:e4:2e:29:dc:a3:90:81:23:7f:fa:
            30:5c:fb:d8:62:7b:96:35:ef:9a:0f:84:49:c4:48:
            97:b5:63:38:91:01:49:f1:42:15:fd:da:84:a6:90:
            4d:2d:05:10:41:cf:06:53:52:80:eb:1b:11:ad:5d:
            63:ed:fe:b1:f7:a7:60:1c:79:b8:88:54:a3:e4:64:
            4d:d3:04:a7:d5:76:17:00:d4:44:19:d6:12:a9:1f:
            aa:2b:ac:73:d6:52:50:92:17:a9:cd:f6:b0:ee:55:
            57:a4:db:82:6e:4f:00:20:6f:6f:f5:b1:72:97:b0:
            c5:3a:88:47:86:c6:e5:dd:fc:91:2f:82:08:05:0c:
            5c:c2:f8:62:92:67:9e:f1:53:24:c0:76:f1:3d:0c:
            50:31:5b:56:26:0a:3b:05:a3:b7:be:f9:ee:a4:82:
            f8:9d:46:ab:a9:dd:b9:04:25:61:58:aa:2a:bb:7c:
            2c:c8:e1:ef:ac:f9:50:e3:64:2e:30:9c:fd:48:26:
            25:7e:75:c0:56:58:10:8d:d7:61:b4:df:f7:ce:bd:
            9c:ef:6f:8b:47:8c:0e:cf:29:ab:eb:33:56:17:99:
            19:ee:30:5f:d9:9d:80:6e:3c:91:05:e6:cd:55:ca:
            25:f2:e3:d9:c8:68:74:1d:9e:4a:e7:53:25:1f:17:
            27:3f:4e:29:c2:19:83:da:4d:8f:b5:6b:5c:de:67:
            4f:01:10:48:84:99:32:c0:e5:e0:8b:9f:eb:4e:18:
            f7:ff:c6:47:b1:47:b8:b2:7f:3c:9c:bd:93:c2:71:
            b3:b4:37:fc:ad:2e:d9:af:2d:2c:f9:de:7f:42:8b:
            39:21:d7:47:8f:18:c4:de:ad:70:0b:11:79:c4:df:
            ef:0f:3a:9a:af:85:4e:95:05:ca:35:9e:6d:93:9b:
            e4:66:23:78:2b:d9:f4:47:e4:fe:29:1e:aa:cb:95:
            66:a2:f2:2a:c3:5a:fa:c0:a0:7d:53:bd:74:37:1d:
            b1:c7:66:67:b7:7b:5f:32:bc:2f:fa:82:0a:12:15:
            2f:41:10:cd:12:70:cc:ee:29:e7:1c:b7:07:d4:28:
            1f:73:3c:15:c0:a2:1d:2b:db:07:57:f7:10:28:c7:
            ed:e4:3a:69:c4:d9:4f:0f:c2:b4:4a:97:2a:2c:b3:
            75:77:5e:1a:21:94:8c:85:fb:0d:5e:95:0f:c8:72:
            59:6c:4f
        generator: 2 (0x2)
```

این دستور از ابزار OpenSSL برای نمایش جزئیات پارامترهای Diffie-Hellman استفاده می‌کند. سوئیچ -in dhparams.pem فایل ورودی که شامل پارامترهای DH است را مشخص می‌کند، -text جزئیات این پارامترها را به صورت متنی نمایش می‌دهد، و -noout مانع از نمایش محتوای خود فایل در خروجی شده و فقط جزئیات را نمایش می‌دهد.
همانطور که در تصویر نشان داده شده، اندازه عدد اول ۴۰۹۶ بیت است و ده بایت آخر نیز با هایلایت نشان داده شده است.

تسک ۵:

سوال ۱:

```
(base) parnian@parnians-MacBook-Pro task05 % shasum -a 256 order.json
2c34b68669427d15f76a1c06ab941e3e6038dacdfb9209455c87519a3ef2c660  order.json
```

دستور shasum -a ۲۵۶ order.json **از ابزار** shasum **برای محاسبه و نمایش هش** SHA-۲۵۶ **یک فایل استفاده می‌کند. سوئیچ** -a ۲۵۶ **الگوریتم هش را به** SHA-۲۵۶ **تنظیم می‌کند، و** order.json **فایل ورودی است که هش آن محاسبه خواهد شد. خروجی این دستور یک رشته هش ۶۴ کاراکتری (هگزادسیمال) است که نشان‌دهنده هش** SHA-۲۵۶ **فایل است.**

سوال ۲:
بعد از تغییر:

```
(base) parnian@parnians-MacBook-Pro task05 % shasum -a 256 order.json
11faeec5edc2a2bad82ab116bbe4df0f4bc6edd96adac7150bb4e6364a238466  order.json
```

سوال ۳:

```
(base) parnian@parnians-MacBook-Pro task05 % python3
Python 3.8.8 (default, Apr 13 2021, 12:59:45)
[Clang 10.0.0 ] :: Anaconda, Inc. on darwin
Type "help", "copyright", "credits" or "license" for more information.
>>> import hmac
>>> import hashlib
>>>
>>> with open('order.txt', 'rb') as file:
...     file_content = file.read()
...
>>> key = b'3RfDFz82'
>>> hmac_result = hmac.new(key, file_content, hashlib.sha256).hexdigest()
>>>
>>> print("HMAC-SHA256:", hmac_result)
HMAC-SHA256: c7e4de386a09ef970300243a70a444ee2a4ca62413aeaeb7097d43d2c5fac89f
```

با استفاده از کد بالا، hmac **را بدست می‌آوریم.**

تسک ۶:

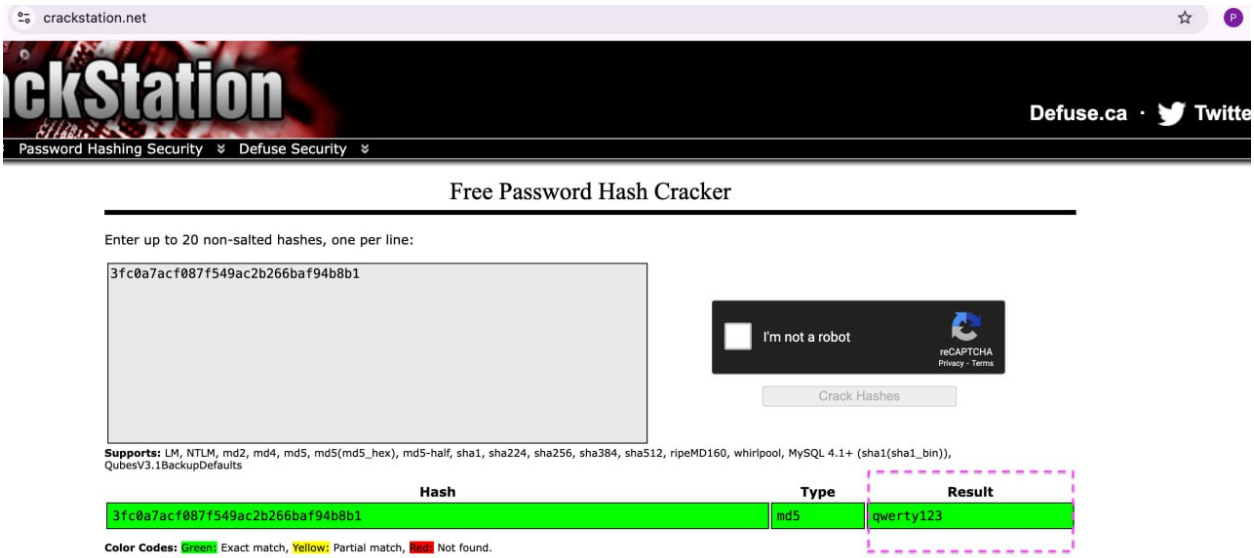
سوال ۱:

```
(base) parnian@parnians-MacBook-Pro task06 % openssl x509 -in cert.pem -text
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number:
            2b:29:0c:2f:b0:52:3a:79:89:1f:82:11:07:bd:9d:84:2a:23:d5:1c
        Signature Algorithm: sha256WithRSAEncryption
        Issuer: C = UK, ST = London, L = London, O = Default Company Ltd
        Validity
            Not Before: Aug 11 11:34:19 2022 GMT
            Not After : Feb 25 11:34:19 2039 GMT
        Subject: C = UK, ST = London, L = London, O = Default Company Ltd
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                RSA Public-Key: (4096 bit)
```

این دستور از ابزار OpenSSL برای نمایش جزئیات یک گواهی X.۵۰۹ استفاده می‌کند. سوئیچ -in cert.pem فایل ورودی گواهی (در اینجا cert.pem) را مشخص می‌کند و -text باعث می‌شود که محتوای گواهی به صورت متنی و با جزئیات (مانند نام صادرکننده، دوره اعتبار، کلید عمومی، و دیگر اطلاعات) نمایش داده شود.
دوره اعتبار و اندازه کلید عمومی در تصویر نشان داده شده‌اند.

تسک ۷:

سوال ۱:



با استفاده از جداول رنگین کمانی که در سایت بالا استفاده می‌شوند میتوانیم گذرواژه اصلی را پیدا کنیم.